CAICT 中国信通院

大数据技术标准推进委员会
Big Data Technology and Standard Committee

redis 中国用户组

内存数据库白皮书

中国信息通信研究院云计算与大数据研究所
CCSA TC601 大数据技术标准推进委员会

2019年6月

版权声明

编委会

牵头编写单位：中国信息通信研究院

参与单位：华为技术有限公司、百度在线网络技术(北京)有限公司、腾讯科技（深圳）有限公司、亚信科技（中国）有限公司、北京极数云舟科技有限公司、中兴通讯股份有限公司、联通大数据有限公司、优刻得科技股份有限公司、阿里云计算有限公司、中移（苏州）软件技术有限公司

编写组成员：

中国信息通信研究院：王妙琼、魏凯、姜春宇、马鹏玮、杨云鹿、田稼丰

华为技术有限公司：黄靖凯

百度在线网络技术(北京)有限公司：廖洪流、曾上游、黄强

腾讯科技（深圳）有限公司：邹鹏、章磊

亚信科技（中国）有限公司：曹升东、张智佳

北京极数云舟科技有限公司：张冬洪

中兴通讯股份有限公司：李忠良

联通大数据有限公司：谢云龙、贾俊峰

优刻得科技股份有限公司：罗成对、王仆

阿里云计算有限公司：姜皓楠、郑明杭、陈琢、朱松

中移（苏州）软件技术有限公司：沈若禺

前　言

随着移动互联网的飞速发展，信息系统的互动性日益增强、用户规模不断攀升，催生出一大批高并发、低时延的新兴应用，这些应用需求对传统系统的性能提出了新的挑战，基于磁盘存储的数据库管理系统由于磁盘读写的速度限制，已经很难满足这类新应用的扩展性和时延要求。

主要依靠内存来存储数据的数据库管理系统，也称为内存数据库，成为了解决高并发、低时延数据管理需求的技术路线。近年来，随着动态随机存储器（DRAM）容量的上升和单位价格的下降，使大量数据在内存中的存储和处理成为可能，Redis、Memcached 等内存数据库管理软件逐渐成熟，应用范围越来越广。未来几年，随着非易失性存储器件（NVM）逐步投入商用，新硬件将会给内存数据库带来更大的发展机遇。

本白皮书阐述了内存数据库的概念，梳理了内存数据库的发展历史和核心属性，分析了在电商、直播和电信行业的典型应用场景，并对主流的内存数据库进行了介绍和对比。白皮书还从技术和管理两个角度提出了产品选型和硬件选型建议，并总结了内存数据库的发展趋势。

本白皮书的编写得到了 Redis 中国用户组的大力支持，在此表示感谢！

目 录

图 表 目 录

一、什么是内存数据库

（一）内存数据库概述

内存数据库又称主存数据库（IMDB/MMDB, In-memory/main memory database），是一种主要依靠内存来存储数据的数据库管理系统[①]。

在数据库技术中，有一类内存优化技术，是在传统的磁盘数据库中，增加内存缓冲池，也就是常说的共享内存技术，其主要目的是最小化磁盘访问。

而内存数据库技术，几乎把整个数据库放进了内存中，相较于传统数据库使用的磁盘读写机制，内存具备更极致的读写速度[②]，性能会比传统的磁盘数据库有数量级的提升。因此内存数据库通常被用于对性能要求较高的场景中。

（二）内存技术的成熟与突破

1.内存技术的成熟

内存器件的容量密度在快速上升。最早期的内存和今天常见的内存条不同，是直接焊接在主板上的内存芯片，容量普遍在64KB以下；1982年之后，随着80286芯片的推出，开始出现30线（pin）256KB的SIMM内存条，被认为是内存领域的开山鼻祖；在80年代末，386和486时代的PC向16位发展，出现了72线的SIMM内存，单条容量可达512KB-2MB；90年代初，EDO DRAM开始盛行，单条容量

① 维基百科： https://en.wikipedia.org/wiki/In-memory_database

② DDR3-1333内存的读写速度约为1GB/s，传统磁盘的读写速度约为150MB/s

在 4MB-16MB；在 1995 年，计算机系统进入图形界面时代，内存技术也发生了重要变革，支持 64 位的 SDRAM 成为一代经典，在性能上有极大提升，容量也达到了 64MB；随后的十几年，内存容量开始稳定地遵循摩尔定律翻倍，持续到 2019 年，DDR3 内存的容量已经可以达到 16GB。

内存器件的单位价格也在逐年快速下降。从 1970 年代至今，内存每兆字节的价格下降了近 9 个数量级，根据 2019 年最新的统计数据，平均花费 3-5 美元就可以购买到 1GB 的内存。

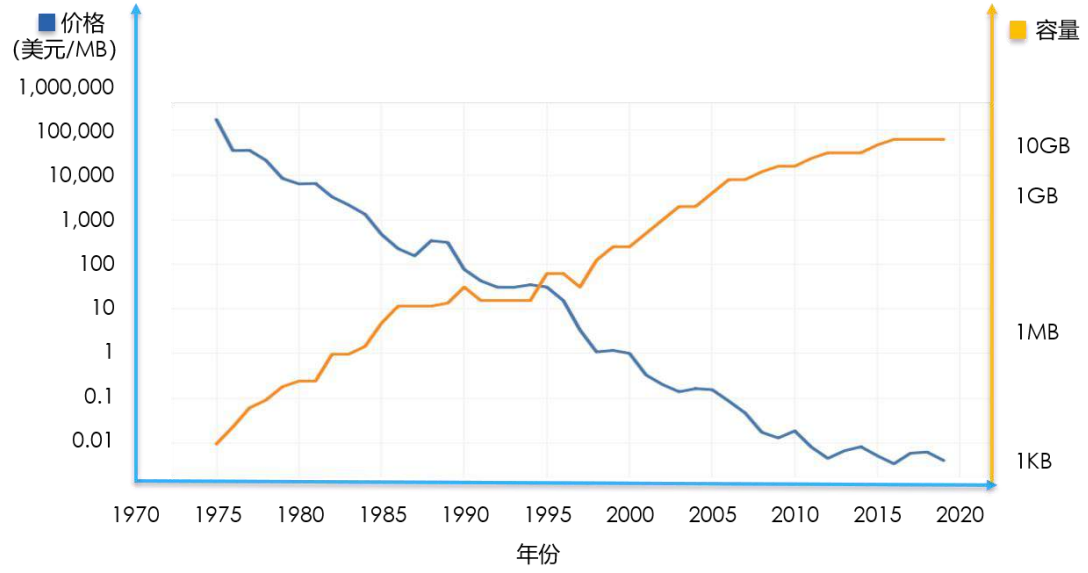

图 1 1970 年代至今的内存价格和容量走势

数据来源：https://jcmit.net/memoryprice.htm，中国信通院整理

内存容量的持续上涨以及价格的下降，使大量数据在内存中进行存储和操作成为可能。

2.内存技术的瓶颈与突破

过去几十年，计算机系统的存储体系结构被设计成如图 2 的金字塔形模型。这样的存储结构利用局部性原理尽量将热数据存储在靠近 CPU 的地方。在传统模式中，内存数据库的所有数据都保存在 DRAM

介质中。虽然 DRAM 的价格已经大幅下降，但在海量数据存储的需求下，内存的成本依然是很大的问题；另外由于 DRAM 属于易失性介质，掉电后所有数据都会丢失，需要额外考虑数据持久化的方案，会极大的限制内存数据库的性能和使用场景。

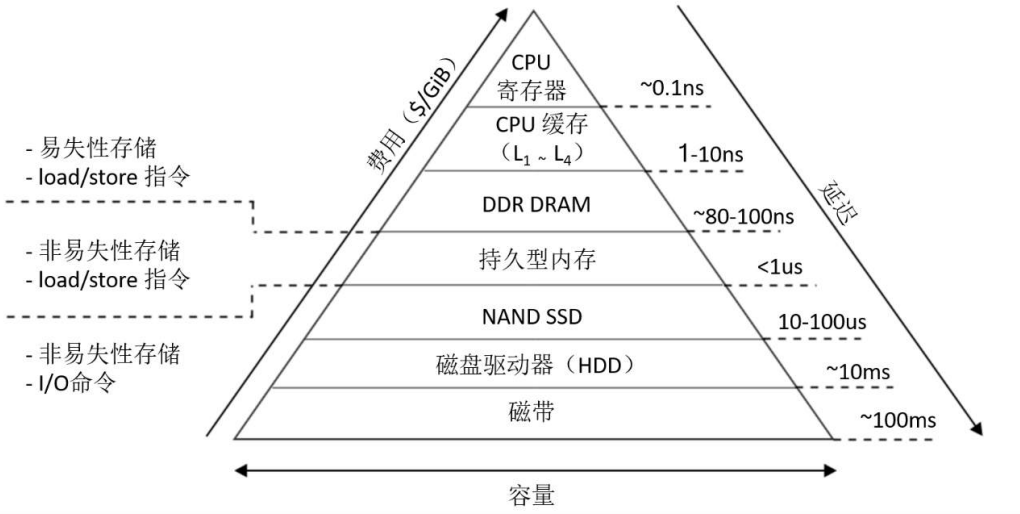

图 2 存储的金字塔模型

针对 DRAM 现存的一些硬件瓶颈，业界已经研发出了持久型内存（PM, Persistent Memory），学术名为存储级内存（SCM, Storage Class Memory），和 DRAM 一样，都是安装在机器主板的内存槽接口中。参考图 2，DDR DRAM 及以上的易失性存储 CPU 可以通过 load/store 指令直接访问，而 NAND SSD 及以下的非易失性存储 CPU 无法直接访问，需要先加载到易失性存储中，可以看出 DRAM 与 SSD 之间存在巨大的性能鸿沟，在访问时延上出现了跳变。而持久型内存位于 DRAM 与 SSD 之间，以 load/store 指令的方式访问并支持数据的持久化，也填补了 DRAM 与 SSD 在时延上存在的鸿沟。相比 DRAM，持久型内存在性能上处于劣势，但容量和价格均占据优势；相比 NAND SSD，持久型内存在性能上处于优势，但容量和价值处于劣势。

持久型内存在 2019 年第一季度已有新产品发布，但尚未大规模商用，随着非易失性存储的规模化使用，克服了硬件上的制约，会使内存数据库的产品能力和应用范围得到大幅度的提升。

（三）内存数据库的发展历程

内存数据库的发展主要经历了雏形期、理论成熟期、市场成长期及高速发展期四个阶段：

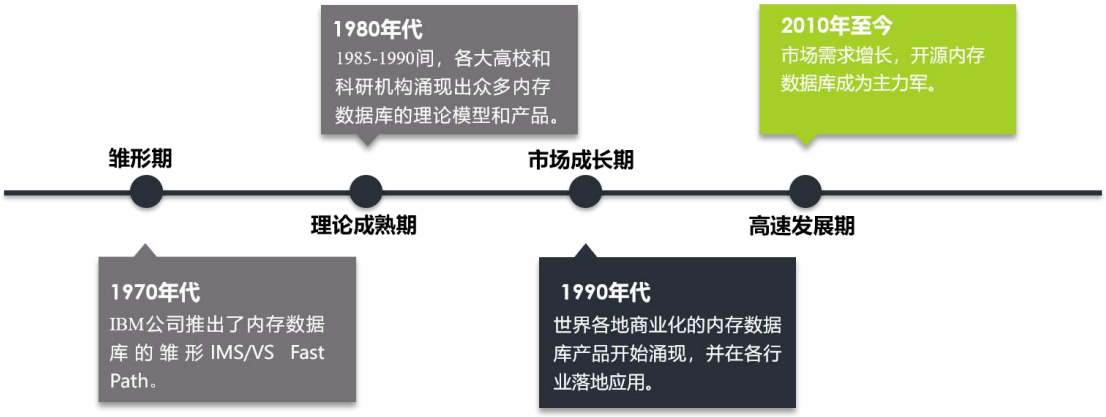

图 3 内存数据库的发展历程

1.雏形期（1970 年代）

在这个时期中，出现了内存数据库的雏形。1969 年 IBM 公司研制了世界上最早的数据库管理系统——基于层次模型的数据库管理系统 IMS，并作为商品化软件投入市场。在设计 IMS 时，IBM 考虑到基于内存的数据管理方法，在 1976 年推出了 IMS/VS Fast Path[2]。Fast Path 提出了数据分层存储的设计理念，将最活跃的数据放在内存中。

2.理论成熟期（1980 年代）

1984 年，D J DeWitt 等人发表了《Implementation Techniques for Main Memory Database Systems》一文。第一次提出了 Main Memory

Database（主存/内存数据库）的概念。并提出使用非易失内存或预提交和组提交技术作为内存数据库的提交处理方案，使用指针实现内存数据库的存取访问。

1985-1990 间，各大高校和科研机构涌现出了众多内存数据库产品。IBM 推出了 IBM 370 上运行的 OBE 内存数据库；威斯康星大学提出了按区双向锁定模式解决内存数据库中的并发控制问题，并设计出 MM-DBMS 内存数据库；贝尔实验室推出了 DALI 内存数据库模型；Southern Methodist 大学设计出 MARS 内存数据库模型；普林斯顿大学设计出 TPK 和 System M 内存数据库。

3.市场成长期（1990 年代）

随着互联网的发展，越来越多的网络应用系统需要能够支持大用户量并发访问、高响应速度的数据库系统，内存数据库市场趋向成熟。半导体技术快速发展，使得半导体内存可以大规模生产，动态随机存取存储器（DRAM）的容量越来越大，价格也越来越低，无疑为计算机内存的不断扩大提供了硬件基础，使得内存数据库的技术可行性逐步成熟。

20 世纪 90 年代，世界各地更多商业化的内存数据库产品开始涌现，并在各行业落地应用。1994 年美国 OSE 公司推出了第一个商业化的内存数据库产品 Polyhedra；随后德国 SoftwareAG 推出了 Tamino Database；日本 UBIT 会社开发出 XDB；韩国 Altibase 公司相继推出 Altibase；奥地利的 QuiLogic 公司推出了 SQL-IMDB；美国 McObject 推出 eXtremeDB。加拿大 Empress 公司推出 EmpressDB。

表 1 1990 年代涌现的商用内存数据库

年份	国家	公司	产品
1994	美国	OSE	Polyhedra
1998	德国	SoftwareAG	Tamino Database
1999	日本	UBIT	XDB 主存数据库产品
1999	韩国	Altibase	Altibase
2000	奥地利	QuiLogic	SQL-IMDB
2001	美国	McObject	eXtremeDB
2001	加拿大	Empress	EmpressDB

4.高速发展期（2010 年至今）

21 世纪初，随着 Web2.0 技术的兴起，互联网进入了一个全新的高速发展期，海量数据的产生使得关系型数据库的存储和快速访问能力面临巨大挑战，越来越多基于非结构化数据模型的应用日趋广泛，这些应用对传统关系型数据库的一些特性并不关注，加上成本更低的开源软件的迅速发展，传统关系型数据库的地位开始受到多方的挑战。

2003 年由 LiveJournal 的 Brad Fitzpatrick 开发完成 Memcached 软件，Memcached 是一个开源的，支持高性能，高并发的分布式内存缓存系统，由 C 语言编写，第一个版本仅 2000 多行代码。但 Memcached 软件不具备持久化功能，数据类型比较少，应用场景比较有限。

2010 年之后，移动互联网进入全面发展期，社交、电商、移动支付、直播、短视频等一大批移动端应用开始涌现，这些应用对系统的高并发、低时延能力提出了极高的要求，也带动了内存数据库的新一轮发展。

2009 年 antirez 发布了 Redis 第一个版本，Redis 是一个开源的使用 ANSI C 语言编写、支持网络、可基于内存亦可持久化的日志型、Key-Value 数据库，并提供多种语言的 API。从 2010 年 3 月 15 日起，

Redis 的开发工作由 VMware 主持。从 2013 年 5 月开始，Redis 的开发由 Pivotal 赞助，至 2015 年 6 月，开始由 Redis Labs 赞助。2018 年 10 月 Redis 发布了 5.0 版本，已经逐渐成为内存数据库的典型代表。

这段时期开源内存数据库开始兴起，如支持关系型存储的 VoltDB；支持键值对存储 NoSQL 数据库 Aerospike；以及数据结构既能支持键值对又能支持关系型的 Apache Ignite，这些开源内存数据库正在成为解决日新月异市场需求的主力军。除了开源数据库之外，SAP 也发布了商业版内存数据库 SAP HANA，是支持列式存储的关系型内存数据库，已经成为内存数据库的典型产品之一。

据 Markets Research Future 在 2019 年 5 月发布的市场研究报告称，预计全球内存数据库市场将以 19%的复合年增长率增长，在 2023 年达到 70 亿美元[1]。

（四）内存数据库的优势与挑战

内存数据库在提供高性能读写能力的同时，也存在由于器件导致的数据易失问题，需要在应用中引起注意。

1.优势：高性能读写

由于省去了磁盘 I/O 的开销，在数据访问的时延上内存型数据库可以达到传统关系型数据库无法达到的微秒级别，单机内存数据库的 QPS 也可以达到 10 万以上，配合上用户态协议栈、内存大页等技术之后，更是可以轻松达到几十万 QPS 的量级，这是传统的关系型数据库很难做到的。

2.挑战：内存数据易失

内存数据库当前主要使用 DRAM 作为存储介质，DRAM 属于掉

电易失性介质，为了保证数据的可靠性，内存数据库需要考虑持久化方案。现阶段主流的键值对内存数据库对于持久化的支持较为薄弱，持久化性能也不如传统数据库。

内存型数据库中克服掉电易失性来保障数据可靠性的方法主要是以下两种：一是每次操作都进行数据持久化，这种方式势必会大幅降低内存数据库的性能；二是按照一定的策略进行操作的持久化，这样可以达到一定程度的优化和缓解，但极端情况下数据丢失的情况仍不可避免。现阶段新型的非易失性存储器件已经发布但尚未规模化商用。相信解决了存储易失性的难题后，内存数据库会具备更多的应用场景。

二、内存数据库的分类及应用场景

（一）内存数据库的分类

主流的内存数据库可分为键值对内存数据库、关系型内存数据库以及其他数据库，用户可根据自身的业务需求选择适合自己的内存数据库类型。

1.键值对内存数据库

键值对（KV, Key-Value）内存数据库指的是一种以键值对为主要存储结构的内存数据库。键值对内存数据库通常按键进行数据存取操作，值通常支持各种数据类型，使用键值存储的数据模型相对简单，更适合要求性能高、计算简单的一些场景。键值对内存数据库的典型代表为 Redis、Memcached 和 Aerospike。

2.关系型内存数据库

关系型内存数据库是一种基于数据关系模型的内存数据库。关系型内存数据库将传统的关系型数据库表搬到内存中，支持通过 SQL 语句的方式实现对内存数据的访问，在实现复杂分析功能的同时，提升数据访问速度。关系型内存数据库的典型代表软件为 Oracle TimesTen、SAP HANA、MemSQL 和 SQLite。

3.其他类型的内存数据库

除键值对内存数据库、关系型内存数据库之外，其他比较小众的内存数据库称为其他内存数据库，比如图内存数据库 RedisGraph 等。

（二）内存数据库的使用场景

1.电商秒杀——键值对内存数据库作为缓冲层的应用

电子商务经过了近 30 年的发展，早已融入了人们的日常生活。为了吸引顾客、推广品牌，各类电商平台都会不定期地举办低价促销和秒杀活动。秒杀活动会对一些特定的商品进行定时、定量售卖，以吸引大量的消费者进行抢购，但又只有极少部分的消费者可以抢单成功。因此，秒杀活动会在较短的时间内产生比平时高数十上百倍的访问流量和下单请求量。根据阿里巴巴的公开数据，在 2017 年双 11 购物狂欢节上，开场 28 秒钟成交额就突破 10 亿，交易峰值 32.5 万/秒，支付峰值 25.6 万/秒，数据库处理峰值 4200 万次/秒。这对数据库在超大并发请求下的稳定性有很高的要求。

一般秒杀活动对系统的压力从秒杀前就会开始并持续到秒杀结束后的一段时间。秒杀前，用户会不断刷新商品页面，页面请求会达到瞬时峰值；秒杀开始的瞬间，大量用户在同一时刻按下秒杀按钮，下单请求会达到瞬时峰值；秒杀结束后，大部分用户还会继续刷单等待退单的机会。

为了在大量业务并发请求下保障系统能够快速稳定地响应用户的请求，可以使用内存型数据库来分流抵挡掉大部分的业务流量，仅仅将需要数据强一致性的请求落到关系型数据库中。该类型的使用场景主要要求内存型数据库拥有足够高的并发读写数量以及并发连接数，根据业务规模的不同并发请求可能会从几十万到上千万，传统的关系型数据库很难满足如此高的性能要求。与此同时，秒杀场景还需要考虑系统的高可用性，通常会使用一主多从的集群模式来防止内存

型数据库出现故障而导致的系统雪崩效应，避免系统由于内存型数据库出现单点故障导致系统整体不可用。

2.视频直播——键值对内存数据库支持高并发+灵活的数据结构

视频直播间作为直播系统对外的表现形式，在整个系统中处于核心地位。除了视频直播窗口外，直播间还会显示在线用户数、实时刷新礼物、评论、点赞、排行榜等信息。直播间的信息显示，流量大、时效性高、互动性强，对系统的并发和性能有着非常高的要求。

除了性能要求，直播系统中对数据结构的灵活性要求非常高，很多场景都适合选择键值对内存数据库，下面以两个案例做介绍。

用户信息。系统中需要维护大量的用户信息，如登录信息、注册信息等。传统的方式是采用关系型数据库存储这些信息，定义一张用户表，用户的属性对应表的列，这种方式的可扩展性很差，用户每增加一个新属性，都需要修改数据库中的表；属于 NoSQL 的键值对内存数据库就能很好的解决这个问题，使用 Key-Value 存储的数据结构非常适合数据的扩展，例如 Redis 提供的 Hashes 数据结构，可以将用户名作为 Key，同时每一行的用户信息是 Hashes 结构内的 Field，用户新增信息时只需要新增一个 Field，可参考图 4 示例。

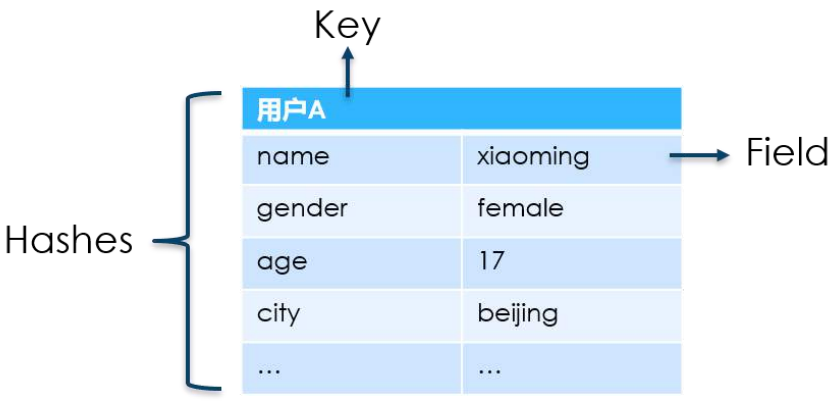

图 4 用户信息使用的数据结构

实时排行。实时排行类信息包含直播间在线用户列表、各种礼物的排行榜、弹幕消息（类似于按时间维度排序的消息排行榜）等，由于用户数量大，使用全表排序压力过大且过程繁琐。Redis 中的有序集合（sorted set）结构就能很好的解决这个问题，集合使用空值散列表（hash table）实现，因此增删改查类操作的时间复杂度都是 O（1）。有序集合中的每个成员都关联一个分数，新增或者修改成员，都会自动进行排序，实时反馈最新的排行。

3.电信计费——关系型内存数据库支持高并发+复杂的数据模型

电信行业计费（Billing）系统是电信支撑系统 BSS/OSS 的核心，其灵活性、响应速度、支撑能力的高低在很大程度上决定了前端业务模式的多样性和客户体验的满意度。

计费系统需存储三户资料、产品订购、账本账单、累积使用量等信息，数据模型复杂、数据量大。以某省移动公司为例，电信计费业务有以下难点：

高并发：每秒需要处理超过 20 万条话单的计费，并发的客户端数超过 1 万个。

低延迟：为使系统中的所有用户都能得到及时响应，在保证高并发、大吞吐量的前提下，系统还需具有低延迟的特性，在线计费服务要求平均响应时间小于 20 毫秒，99%响应时间小于 100 毫秒。

高可用：计费系统作为运营商的生产系统，一旦系统故障，将影响终端客户的业务使用（包括语音服务、数据服务），所以需要计费系统（包括关系型内存数据库）具有电信级的高可用能力，即 99.999%的可用性。

数据模型复杂：话单的计费处理涉及到多种数据：单次费用明细、

号码级别累计使用量、不同时段累计量、账户级别累计量等的访问与修改，这些数据的访问与修改需要 SQL 兼容性及 ACID 保证，以及安全的读写隔离级别。

计费系统需要用到的数据分组包括通话详单、用户属性（包括用户号码、资费方案、入网时间等）、资费参数（包括字头、费率、折扣等），计费系统以通话详单为中心，查询相关的用户属性、资费参数，根据查询结果，计算话单费用[3]。从应用特点来看，一般用于实时计费业务会选择 SQL 兼容性较高的关系型内存数据库，一是关系型数据库更适合数据关联性强、数据模型复杂的场景，一定程度上比 NoSQL 数据库更能保证 ACID 事务；二是相较于传统计费系统使用的磁盘存储批处理方式，内存数据库可以实现数据在内存中的实时累计和查询，免去了磁盘数据库和内存间周期性同步的时间差，能极大地缩短查询的响应时间。

三、内存数据库的选型建议

（一）内存数据库产品现状

DB-Engines Ranking 是公认较权威的数据库排行，我们选取了其中最为活跃的 10 款典型内存数据库进行对比。开源产品中，Redis 和 Memcached 是最受欢迎的两款键值对内存数据库；而 SQLite 是最受欢迎的关系型内存数据库。表中大部分的关系型内存数据库为商用数据库，其中热度最高的是 SAP HANA。早在 1995 年就发布第一版的 Oracle TimesTen 仍然在榜上活跃；2014 年新发布 Apache Ignite 兼容键值和关系型数据结构，热度正稳步攀升。事务支持方面，大部分的关系型内存数据库称可以支持 ACID，但都需要在性能上作出妥协。具体可参考表 2 中的信息。

表 2 10 款典型内存数据库对比

数据库名称	数据结构	起始年份	商用/开源（License）	ACID 支持情况
Redis	键值对	2009 年	开源（BSD）	不支持
Memcached	键值对	2003 年	开源（BSD）	不支持
Aerospike	键值对	2012 年	开源（AGPL）	只支持原子性
Apache Ignite	键值对/关系型	2014 年	开源（Apache2.0）	支持
SAP HANA	关系型	2010 年	商用	支持
Oracle TimesTen	关系型	1995 年	商用	支持
VoltDB	关系型	2010 年	商用/开源（GPL）	支持
MemSQL	关系型	2013 年	商用	只支持隔离性
SQLite	关系型	2000 年	开源（Public domain）	支持
eXtremeDB	关系型	2001 年	商用	支持

（二）内存数据库选型建议

技术服务于业务，内存数据库的选型应首先遵循业务场景的需求。业务特性决定了数据的应用特性，包括数据量、并发度、读写特性、一致性、响应时间、操作复杂度、业务连续性等要求，对应数据库的一致性、容错性、扩展性、安全性等技术要求。在做内存数据库的选型前，建议先梳理业务需求并进行量化；再将核心数据应用特性映射成数据库技术要求；最后按筛选出的技术要求进行选型。

1.技术因素

按照技术要求进行内存数据库选型时，可主要考察业务的性能、一致性要求和 SQL 兼容性三个因素。具体选型思路可参考图 5。

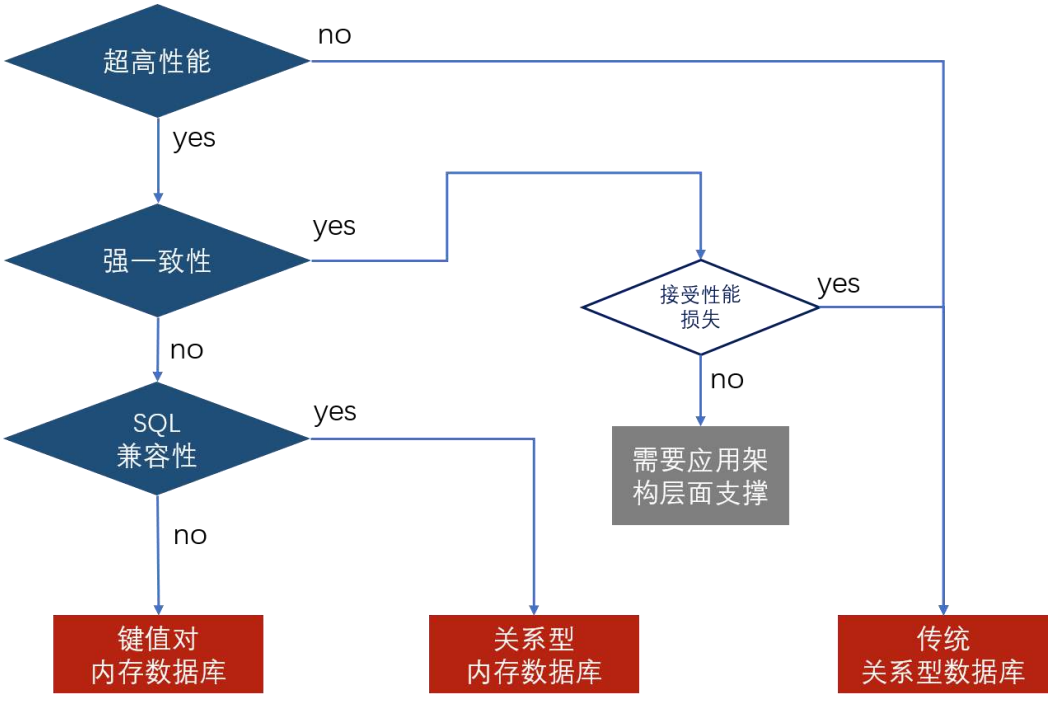

图 5 内存数据库选型建议

业务是否有很高的性能要求？一般有高并发、低时延读写要求的业务，如游戏实时排行、直播粉丝关注等，建议选择内存数据库。

业务数据是否要求强一致性？如果业务对数据的可靠性和一致

性要求较高、需要 ACID 级别的事务支持，则建议使用 MySQL 等传统的关系型数据库。但需要注意的是，强一致性的要求会对数据库的性能造成一定的影响；如果需要兼具高性能和强一致性，则需要在应用架构层面进行优化，单靠数据库的能力还无法实现。

数据处理是否要求 SQL 兼容性？在高性能要求的场景下，业务中如果数据结构固定、有复杂的关联计算要求，或是需要 SQL 语法支持的情况，建议使用关系型内存数据库；对于数据结构多变、扩展性要求高、数据模型和操作简单的场景，建议使用键值对内存数据库。

除了这三条考察指标，还可以结合数据容量、成本、扩展性、可维护性等需求进行综合考量。

2.非技术因素

上述选型方法主要考量的是技术因素，除此以外还可以结合实际情况，引入一些其他维度的考量，进行综合评估，最终挑选出适合的产品。包括但不限于以下维度：

1）生态成熟度。指数据库产品的状态，包括各种配套工具、技术架构成熟度、代码质量、开发模式、社区建设、商业支持服务、版权协议等；

2）应用架构适配度。指应用架构对数据库架构的兼容性、以及适配改造友好度，包括技术架构适配、开发语言适配等；

3）团队适应度。指开发团队、维护团队对数据库的熟悉程度、偏好程度、学习成本以及配套运维工具等。

（三）硬件选型建议

内存数据库是为高性能而存在的，适合才是最好的，因此在硬件选型上建议根据业务情况综合考虑成本和收益。

1.如何选择存储

作为内存数据库，硬件选型的时候，优先考虑的就是大内存，比如：256GB 或 512GB。如果在技术选型的时候，业务上要求有持久化的功能，那么在硬件选型的时候需要考虑到磁盘的 IOPS（Input/Output Operations Per Second），比如选择了 Redis，需要做 RDB+AOF 的持久化，那么硬盘最好是 SSD 或者 PCI-e（当前基于 NVM 协议的 SSD 比较适合）。

如果有足够的预算，并且业务对内存数据的访问量不是非常大，响应时间没有那么高，那么也可以考虑用 Persistent Memory（这种新技术的性能是物理内存性能的 70-80%）。

还有一种特别的使用场景，那就是冷热数据分离，热数据存储在内存中，冷数据存储在磁盘上，这种常常会涉及到冷热数据的相互交换，那么这种情况，就需要磁盘要有足够高的性能，PCI-e 最佳。

2.如何选择网络

千兆网卡即可满足大部分需求，如果预算充足并希望达到极致的性能，可选择万兆交换机和万兆网卡。

3.如何选择 CPU

基于 X86 架构的 CPU 单核心性能更高，能适配大部分的内存数据库产品，成本也相应更高；ARM 架构的 CPU 成本更低廉，但可以适配的内存数据库产品有限。

四、内存数据库技术演进趋势

（一）内存数据库和传统数据库混合使用将成为主要模式

随着业务的增长,很多内存数据库的存储容量已经达到TB级别，内存数据库海量存储成为刚需，同时还要兼具性能优势，成本和扩展性成为了两大难题。

数据分离是现阶段实现这种愿景的一种可行方案。一般会按照数据访问的冷热作为分离的判定条件，将冷数据从内存中转移，选择性能和成本较低的 SSD 或磁盘来存储。这种混合内存和磁盘的数据库会将所有的热数据保留在内存中,业务根据自己需求调整内存和磁盘的比例，可以在性能和成本之间取得一种平衡。数据分离需要设计合适的机制来处理冷热数据识别、冷热数据交换。

（二）软硬件深度整合为内存数据库开辟新的技术方向

随着业务逻辑越来越复杂，业务系统越来越庞大，业务对内存数据库的性能和容量的需求也亦发强烈。在内存数据库的规模达到一定程度后，传统的“通用硬件+通用软件”的整合模式已经达到瓶颈，无法再进一步满足业务需求,所以必然会发展出软硬件深度整合的混合存储数据库系统，硬件涉及网络、存储、内存、处理器四个主要的子系统。

1.网络子系统

网络子系统是业务和内存数据库交互的“高速公路”，网络子系统的性能直接影响了内存数据库的输入输出能力。

在传统的“通用硬件+通用软件”的模式中，网卡硬件负责数据收发，IP 协议和 TCP 协议则由操作系统内核协议栈进行处理，内存

数据库进程再通过用户态和内核态数据拷贝的方式与操作系统内核通信。这种模式存在“操作系统通用 TCP/IP 协议栈过于笨重、用户态和内核态数据拷贝效率低下”等缺点，导致网卡的硬件能力无法发挥到极致。

在新的软硬件整合模式中，不再使用操作系统内核的 TCP/IP 协议栈，而是将 IP 协议卸载到智能网卡中，TCP 协议则卸载到内存数据库的用户态进程中。在这种模式下，网卡硬件内置了 IP 协议栈，网卡除了负责网络数据包的收发，同时也负责 IP 协议的处理，网卡将处理后的 IP 报文直接交给用户态的内存数据库进程进行 TCP 协议处理。经内部测试，采用新的软硬件整合模式，网络子系统的性能相对传统模式至少可提升 300%以上。

2.存储子系统

存储子系统负责内存数据库的持久化数据的存储和加载。对于需要持久化数据的场景，存储子系统的能力直接影响内存数据库的数据写入能力。

在传统的“通用硬件+通用软件”的模式中，存储硬件由操作系统内核负责管理，内存数据库的进程则通过用户态和内核态的数据拷贝方式与操作系统内核进行通信。这种模式存在“操作系统内核的通用文件系统过于笨重，用户态和内核态数据拷贝效率低下”的缺点，导致存储子系统整体性能不高。

在新的软硬件整合模式中，存储硬件不再由操作系统内核负责管理，而是直接由内存数据库的用户态进程进行管理，内存数据库的进程通过用户态存储驱动直接与存储硬件进行交互。通过将存储硬件卸载到用户态来管理，存储系统的能力的可得到极大的提升。如果存储

硬件再采用专为键值对内存数据库优化的 SSD，则存储性能还能得到更大的飞跃。

3.内存子系统

内存子系统软硬件整合的主要目的是为了增大内存数据库的容量并降低成本。

在传统模式中，内存数据库的所有数据都保存在 DRAM 介质中。虽然近几年 DRAM 技术在业界的大力发展下，容量已大幅提升，成本也相对下降，但依然还是解决不了内存数据库容量巨大时，DRAM 成本高居不下的问题。另外由于 DRAM 属于易失性介质，掉电后所有数据都会丢失，恢复后则需要从存储子系统重新加载大量数据，对存储子系统也造成了巨大的压力，并且恢复时间也相对较长。

针对 DRAM 成本高、掉电丢失数据的缺点，业界已经研发出新的非易失性存储器件。这种新存储器件和 DRAM 一样，都是安装在机器主板的内存槽接口中，但同等容量的成本却只有 DRAM 的 50%左右，性能却能达到 DRAM 的 70%~90%；而且新存储介质是非易失性的，即掉电后不会丢失数据，系统恢复后不必要从存储子系统加载大量数据，可实现掉电后秒级恢复数据。随着非易失性存储的成熟和规模化商用，可能会使数据库对于 SSD、磁盘存储等持久性存储的依赖带来颠覆性的变化。

4.处理器子系统

处理器子系统是内存数据库系统执行代码逻辑的关键所在，随着“摩尔定律”的失效，CPU 的单核计算性能在最近几年已无极大飞跃，难于提升内存数据库的单核计算性能；而且由于功耗、发热等技术限制，多核技术也无法实现 CPU 核心数的持续增加。如果还是采用传

统的通用 CPU 模式，所有算法逻辑必须通过软件编码的方式来实现，这种“通用 CPU+软件算法”的模式对内存数据库用于大数据和人工智能等场景已经捉襟见肘。

针对“通用 CPU+软件算法”的缺点和瓶颈，业界已经在针对各种场景开发专用的 FPGA 芯片，特别是人工智能场景中，各种 AI 智能芯片已经发布并开始在产品中成功应用。针对内存数据库和大数据、人工智能等相结合的 FPGA 芯片当前业界也已经开始研究和开发。将 FPGA 芯片应用到内存数据库系统后，内存数据库的计算能力将得到极大的提升和飞跃。

（三）协议创新将进一步提升分布式内存数据库的一致性能力

在金融和涉及交易的系统中，传统上都是使用关系型数据库来存储数据，键值对内存数据库相比关系型数据库，使用更简单，扩展更方便，一部分用户正在尝试使用键值对内存数据库来存储这些重要数据。一般基于键值对的内存数据库都是异步地进行主从复制，主从副本之间达到数据完全一致存在延迟。在主从副本达到完全一致之前，如果发生主从切换，一部分的写入数据就有可能丢失。

传统的方案是在每一次写入请求时检查异步复制的进度，只有主从同步达到一致，才允许写入，这无疑会损失很大一部分的性能。在有多个从副本的情况下，需要所有的副本都达到一致才允许下一次写入。检查多少个副本取决于对一致性要求的高低和对性能下降的容忍度。

对于同地域有强一致性需求的业务，又需要保证性能不会大幅下

降，使用 Raft 协议是一大趋势。Raft 协议保证数据在大部分节点都写入成功的情况下，就确认一致性，这样可以在最大程度保有性能的同时，减少单个节点处理慢或者节点故障影响到系统的可用性。

而对于社交、视频直播、电商以及游戏等需要跨数据中心进行数据同步的场景，可以使用 CRDT（Conflict-Free Replicated Data Type）的最终一致性方案。跨地域同步时，数据会在多个地域产生，这样会导致数据出现冲突，CRDT 是各种基础数据结构最终一致算法的理论总结，能根据一定的规则自动合并，解决冲突，达到强最终一致的效果。

（四）与容器技术结合为内存数据库提供更强的弹性扩展能力

随着互联网的发展及企业业务规模的扩张，很多业务初期规划的系统规模已经不能承载飞速发展的业务需求，所以弹性可扩展能力也是内存数据库发展中的重点。

近年来容器技术的成熟和普及为弹性扩展提供了可行的解决方案，容器的使用会带来很多优势：

更小的资源消耗：与之前广泛使用的虚拟机相比，容器本身所消耗的资源更少，节省下来的资源可以承担更多的内存数据库的计算量，在相对固定的总资源基础上，可为扩展数据库提供更多的空间。

更高的资源利用率：在一个宿主机上，可以运行成百上千个容器，如果把每个容器都想象成一个内存型数据库，则能极大提升宿主机的资源使用率，当数据库需要扩展时，可以很容易的找到可部署的宿主机资源，为创建更多数据库实例提供了可能。

易迁移：考虑到内存数据库的扩展需求，需要在不同环境做快速地迁移，在容器的帮助下，我们不需要在意宿主机的环境，只需要制作好镜像就能实现快速安全的迁移。

快速部署：弹性扩展很重要的一个指标就是新增节点的速度，容器相比于传统的虚拟机在这方面具备了很大的优势，一般在几秒内就能完成部署，非常适合一些紧急的扩容场景。

管理自动化：发展飞速的容器管理技术，比如 Kubernetes，把容器的自动化管理推上了一个新的高度，也为内存数据库的自动化部署和弹性扩容提供了可能，可以通过预先部署的监控自动触发弹性扩容的程序，实现自动化的弹性扩缩容。

五、 总结与展望

从 1976 年诞生了内存数据库的第一个雏形开始，其经历了理论成熟期和市场成长期，2010 年之后，随着移动互联网飞速发展，高并发、低时延的应用需求催生出一批开源和商业内存数据库。

内存数据库由于省去了磁盘读写的开销，在性能上会比使用磁盘存储的传统的数据库有数量级的提升，但由于现阶段仍在使用掉电易失的 DRAM 器件，为了保证数据的可靠性，内存数据库需要考虑持久化方案。随着未来非易失性存储器件的发展，将为内存数据库的数据持久化问题带来新的解决方案。

内存数据库产品选型时，需要关注业务对于性能、一致性和 SQL 支持度等的需求，除了这些技术因素，还可以引入其他维度的考量，如生态成熟度、应用架构适配度、团队适应度等。此外，因为使用内存数据库的场景多关注性能，所以选择合适的硬件搭配也很重要。

对于内存数据库的技术演进趋势，我们有以下几点观察：一是内存数据库和传统数据库混合使用将成为主要模式；二是软硬件深度整合能为内存数据库开辟新的技术方向；三是协议创新将进一步提升分布式内存数据库的一致性能力；四是与容器技术结合能为内存数据库提供更强的弹性扩展能力。

非易失性存储器件的发展进入商用前夜，将对传统的存储模式带来颠覆性的变化。在市场需求和新器件的双重驱动下，内存数据库将能适用更广泛的应用场景并获得更多的发展机会。

参考文献

[1] In-Memory Database Market Research Report - Global Forecast to 2023[R].2019

[2] Gawlick, Dieter & Kinkade, David. (1985). Varieties of concurrency control in IMS/VS fast path. IEEE Database Eng. Bull.. 8. 3-10.

[3] 武振宇. 内存数据库及其在实时计费系统中的应用[J]. 电信工程技术与标准化, 2012, 25(3):62-65.

[4] Levandoski J . Modern main-memory database systems[M]. VLDB Endowment, 2016.

[5] H. Garcia-Molina and K. Salem, "Main memory database systems: an overview," in IEEE Transactions on Knowledge and Data Engineering, vol. 4, no. 6, pp. 509-516, Dec. 1992.

[6] The Forrester Wave™: In-Memory Databases, Q1 2017 In-Memory Databases Are Driving Next-Generation Workloads And Use Cases[R].2017

附件：缩略语

缩略语	英文	中文
ACID	Atomicity、Consistency、Isolation、Durability	原子性、一致性、隔离性、持久性
CAP	Consistency、Availability、Partition tolerance	一致性、可用性、分区容错性
CPU	Central Processing Unit	中央处理器
CRDT	Conflict-Free Replicated Data Type	无冲突复制数据类型
DB	Data Base	数据库
DDR DRAM	Double Data Rate DRAM	双倍速率同步动态随机存储器
DRAM	Dynamic Ramdom Access Memory	动态随机存取存储器
IOPS	Input/Output Operations Per Second	每秒读写次数
IP	Internet Protocol Address	互联网协议地址
KV	Key-Value	键值对
NAND SSD	NAND Solid State Drive	基于闪存的固态存储
NoSQL	Not Only Structured Query Language	非结构化查询语言
NVM	Non-volatile Memory	非易失物理存储介质
PM	Persistent Memory	持久型内存
SCM	Storage Class Memory	存储级内存
SQL	Structured Query Language	结构化查询语言
SSD	Solid State Drive	固态存储
TCP	Transmission Control Protocol	传输控制协议

中国信息通信研究院

中国信息通信研究院云计算与大数据研究所

地址：北京市海淀区花园北路 52 号

邮政编码：100191

联系电话：010-62304616

邮箱：wangmiaoqiong@caict.ac.cn

网址：www.caict.ac.cn